Machine-learning approach for operating electron beam at KEK electron/positron injector linac

Gaku Mitsuka,[1,2,*] Shinnosuke Kato,[3] Naoko Iida,[1,2] Takuya Natsui,[1,2] and Masanori Satoh[1,2]
[1]*KEK, Oho, Tsukuba, Ibaraki 305-0801, Japan*
[2]*SOKENDAI, Shonan Village, Hayama, Kanagawa 240-0193, Japan*
[3]*The University of Tokyo, Bunkyo, Tokyo 113-0033, Japan*

(Received 28 January 2024; accepted 8 August 2024; published 26 August 2024)

In current accelerators, numerous parameters and monitored values are to be adjusted and evaluated, respectively. In addition, fine adjustments are required to achieve the target performance. Therefore, the conventional accelerator-operation method, in which experts manually adjust the parameters, is reaching its limits. We are currently investigating the use of machine learning for accelerator tuning as an alternative to expert-based tuning. In recent years, machine-learning algorithms have progressed significantly in terms of speed, sensitivity, and application range. In addition, various libraries are available from different vendors and are relatively easy to use. Herein, we report the results of electron-beam tuning experiments using Bayesian optimization, a tree-structured Parzen estimator, and a covariance matrix-adaptation evolution strategy. Beam-tuning experiments are performed at the KEK e^-/e^+ injector Linac to maximize the electron-beam charge and reduce the energy-dispersion function. In each case, the performance achieved is comparable to that of a skilled expert.

DOI: 10.1103/PhysRevAccelBeams.27.084601

I. INTRODUCTION

To improve or maintain the high performance of modern accelerators, dozens or even hundreds of parameters must be optimized to accommodate the volatile conditions. Values monitored to determine the success or failure of the optimization include those of the beam orbit, beam charge, energy-dispersion function, emittance, and charge loss in each accelerator sector. Hundreds of values are monitored. Determining the operating parameters, such as the magnetic field, based solely on the beam dynamics is typically challenging. For example, the KEK e^-/e^+ injector Linac (referred to as the KEK Linac) has no monitors to diagnose the beam energy in each sector, and the energy gains of individual rf cavities are accurately determined only occasionally. In addition, changes in the environmental temperature affect the rf system and set energy drifts. Therefore, the actual operation requires beam-parameter optimization while the beam conditions are monitored. Hitherto, operation experts have optimized the beam parameters based on their knowledge and experience. Complex and sensitive accelerator operations, such live optimizations based on expert inputs, may be time consuming to reproduce, even if the results satisfy the required criteria. In particular, in the case of Linac accelerating a beam in a single pass, a self-feedback mechanism does not exist, unlike the ring where the beam orbits. Thus the beam condition cannot be reproduced easily even if the same operating parameters are set.

Accelerator tuning using machine learning has recently garnered attention as an alternative to expert-dependent optimization (e.g., see Refs. [1,2]). Machine learning has progressed significantly in terms of speed, sensitivity, and application range since the 2010s. Various libraries are available from different vendors and are relatively easy to use. When applying machine learning to accelerator tuning, we must consider that the accelerator conditions at the time of tuning are not always similar to those at past days. For example, environmental changes, such as temperature drift occurring even day and night, often affect the beam orbit or intensity. We also consider that multiple optimization steps are required to improve the prediction accuracy. Nevertheless, in accelerator tuning, where the cost of a single step cannot be disregarded, one should refrain from performing numerous steps. Owing to these demands, the objective of this study is to realize accelerator tuning using black-box parameter-optimization algorithms.

In this paper, we evaluated Bayesian optimization [3] for actual beam tuning to determine whether the target performance was achieved, the number of steps required for the specified parameters, and whether multiple goals (e.g., beam charge vs dispersion function) were achieved

*Contact author: gaku.mitsuka@kek.jp

simultaneously. Bayesian optimization is a parameter-optimization algorithm based on the Gaussian process [4] and Bayesian decision theory [5]. In addition to Bayesian optimization, we evaluated other parameter-optimization algorithms, i.e., the tree-structured Parzen estimator (TPE) [6,7] and the covariance matrix-adaptation evolution strategy (CMA-ES) [8], by comparing their characteristics and tuning results. We performed a beam-tuning experiment at the KEK Linac in June 2023 using a parameter-optimization library name OPTUNA [9]. OPTUNA internally implements not only the BoTorch [10] (Bayesian optimization), TPE, and CMA-ES algorithms but also other single- and multiobjective optimization algorithms. The results of beam tuning using various optimization algorithms provide essential guidelines for future machine-learning applications associated with particle accelerators.

The remainder of this paper is organized as follows: Sec. II describes the flow of a single-objective optimization using Bayesian optimization, the TPE, and the CMA-ES. Section III reports the experimental results of the single-objective optimization performed at the KEK Linac. Section IV describes the flow of multiobjective optimization. Section V details an actual beam experiment performed at the KEK Linac. Finally, Sec. VI concludes this paper.

II. SINGLE-OBJECTIVE OPTIMIZATION

For simplicity, we define *parameter optimization* as the process of identifying optimal *parameter* values (e.g., applied currents of steering magnetic coils) to *minimize* the *objective function* (e.g., dispersion function). If the parameters are to be optimized by *maximizing* the objective function (e.g., the beam charge), then the discussion in this section can be applied by reversing the sign of the evaluated value.

A. Bayesian optimization

Bayesian optimization is an iterative strategy for the global optimization of black-box functions. It builds a surrogate model of the black-box objective function using a stochastic process known as the Gaussian process [4]. The optimization follows an acquisition function to guide the parameter space exploration. Acquisition functions can be interpreted in the framework of Bayesian decision theory [5]. A detailed description of the Gaussian processes is provided in Ref. [4]. In this study, we used the BoTorch algorithm [10] via OPTUNA [9] for Bayesian optimization.

In Bayesian optimization, the objective function $f(x)$ for each putative input location x is assumed to adhere to a Gaussian process $\mathcal{GP}(0,k)$ with a mean function of 0 and a covariance function k. The covariance function is similarly known as the kernel function, and the Matern kernel function is used in this study—the Matern kernel function is chosen as the default kernel function in BoTorch internally implemented in OPTUNA:

$$k_\nu(x,x') = \frac{2^{1-\nu}}{\Gamma(\nu)}(\sqrt{2\nu}d)^\nu K_\nu(\sqrt{2\nu}d)$$
$$d = (x-x')^T\Theta^2(x-x'). \tag{1}$$

Here, K_ν is the modified Bessel function of the second kind. The parameter ν determines the smoothness of the function, and we use $\nu = 5/2$ in this study. The lengthscale parameter Θ scales the kernel function and is estimated from data. The automatic relevance determination technique is employed in this study to set the lengthscale parameter Θ very long for some irrelevant dimensions, resulting in a covariance function effectively removing such dimensions [3,4].

In a Gaussian process, the observed value can be written as $y = f(x) + \varepsilon$. A noise ε is assumed adhere to $\varepsilon \sim \mathcal{N}(0,\sigma^2)$, where a constant variance σ^2 is independent of the location x and is inferred in the algorithm. Under these assumptions, we obtain the following probability model between location x and observed value y:

$$p(y|x,\mathcal{H}_t) = \mathcal{N}(\mu_t(x,\mathcal{H}_t),\sigma_t(x,\mathcal{H}_t)^2). \tag{2}$$

$\mathcal{H}_t = \{(x_i,y_i)\}_{i=1}^{t-1}$ is the history of pairs of location x and observed value y up to the $t-1$th step. For the specific forms of μ_t and σ_t, refer to Ref. [4].

The acquisition function in Bayesian optimization is defined as a real-valued function $\alpha(x,\mathcal{H}_t)$ in the space of objective function $\mathcal{X}$. In each step, the location x that maximizes the acquisition function α is selected, which is then input to the objective function $f(x)$ to obtain the observed value y. The expected improvement (EI) is applied in the acquisition function used in this study:

$$x_t \in \arg\max_{x'\in\mathcal{X}} \alpha(x';\mathcal{H}_t)$$
$$\alpha_{\mathrm{EI}}(x;\mathcal{H}_t) = \int \max(y_t^* - y, 0)\, p(y|x,\mathcal{H}_t)\, dy. \tag{3}$$

Specifically, the EI represents the expected value for the conditional probability distribution $p(y|x,\mathcal{H}_t)$ and the extent by which the observed value y improves from the currently obtained minimum y value (denoted y_t^*) based on a location x selected; and y_t^* is a constant determined based on the history $\mathcal{H}_t$.

B. TPE

Unlike Bayesian optimization with Gaussian processes, which directly models $p(y|x,\mathcal{H}_t)$, as shown in Eq. (2), the TPE [6] models $p(x|y,\mathcal{H}_t)$ and $p(y)$ and uses them to calculate the EI. The conditional probability distribution can be modeled using two densities, as follows:

$$p(x|y,\mathcal{H}_t) = \begin{cases} l(x) & (y < y_t^*) \\ g(x) & (y \geq y_t^*) \end{cases}. \tag{4}$$

The distribution exhibited by x when the observed value y is lower than y_t^* is denoted as $l(x)$, and the distribution when y is larger than y_t^* as $g(x)$. The two distributions in Eq. (4) are obtained using kernel density estimators [7]. y_t^* shall be provided to satisfy the probability $\gamma = p(y < y_t^*|\mathcal{H}_t)$ for the predefined threshold γ $(0 < \gamma < 1)$.

Using Eq. (4), we can obtain the EI from the TPE:

$$\begin{aligned}\alpha_{\mathrm{EI}}(x;\mathcal{H}_t) &= \int_{-\infty}^{\infty} \max(y_t^* - y, 0)\, p(y|x,\mathcal{H}_t)\, dy \\ &= \int_{-\infty}^{y_t^*} (y_t^* - y)\, \frac{p(x|y,\mathcal{H}_t)\, p(y|\mathcal{H}_t)}{p(x|\mathcal{H}_t)}\, dy \\ &= \frac{l(x)}{p(x|\mathcal{H}_t)} \int_{-\infty}^{y_t^*} (y_t^* - y)\, p(y|\mathcal{H}_t)\, dy, \end{aligned} \tag{5}$$

where the normalization in the denominator is written as

$$\begin{aligned} p(x|\mathcal{H}_t) &= \int p(x|y,\mathcal{H}_t)\, p(y|\mathcal{H}_t) \\ &= \gamma l(x) + (1-\gamma) g(x). \end{aligned} \tag{6}$$

The final integral in Eq. (5) is independent of x. Thus the specific form of $p(y|\mathcal{H}_t)$ need not be considered when maximizing $\alpha_{\mathrm{EI}}(x;\mathcal{H}_t)$. Finally, we approximate Eq. (5) as

$$\begin{aligned}\alpha_{\mathrm{EI}}(x;\mathcal{H}_t) &= \frac{l(x)}{p(x|\mathcal{H}_t)} \int_{-\infty}^{y_t^*} (y_t^* - y)\, p(y|\mathcal{H}_t)\, dy \\ &\propto \frac{l(x)}{\gamma l(x) + (1-\gamma) g(x)} \\ &= \left(\gamma + \frac{g(x)}{l(x)}(1-\gamma)\right)^{-1}. \end{aligned} \tag{7}$$

Eq. (7) indicates that the x value maximizing $\alpha_{\mathrm{EI}}(x;\mathcal{H}_t)$ is the location that minimizes the density ratio $g(x)/l(x)$. As detailed in Ref. [7], smaller γ value leads to more exploration and less exploitation, and we use $\gamma = 0.1$ in this study.

C. CMA-ES

The CMA-ES [8] is an evolutionary computational algorithm for continuous optimization problems. In each iteration (*generation*, denoted as g), new candidate solutions (*individuals*, denoted as x) are generated following the multivariate normal distribution determined by the parental individuals. The set of individuals is known as the *population*, and the *population size* in each generation is denoted by λ. The algorithm comprises the following steps: (i) Generate λ candidate solutions (individuals) based on the multivariate normal distribution $\mathcal{N}(m, \sigma^2 C)$ and calculate the objective function for each individual. (ii) Among the λ individuals generated, extract $\mu(\mu < \lambda)$ individuals with the highest-ranking objective functions and update the mean vector m by multiplying the μ individuals by their weights. (iii) Update the variance parameters (i.e., σ and C) of the multivariate normal distribution based on the isotropic and anisotropic evolution paths. (iv) Repeat steps 1–3.

Step 1: The kth individual $x_k^{(g+1)}$ in generation $g+1$ can be determined based on the multivariate normal distribution $x_k^{(g+1)} \sim \mathcal{N}(m^{(g)}, (\sigma^{(g)})^2 C^{(g)})$. The mean vector $m^{(g)}$, step size $\sigma^{(g)}$, and covariance matrix $C^{(g)}$ are parameters constructed from the generation g. The objective function for $x_k^{(g+1)}$ is denoted as $f(x_k^{(g+1)})$.

Step 2: Define $x_{i:\lambda}^{(g)}$ as the λ individuals in generation g sorted in the ascending order of the corresponding objective functions. The mean vector $m^{(g)}$ is written as

$$m^{(g)} = \sum_{i=1}^{\mu} w_i x_{i:\lambda}^{(g)}, \tag{8}$$

where w_i is the weight of each individual and satisfies the relation:

$$\sum_{i=1}^{\mu} w_i = 1 \qquad (w_1 \geq w_2 \geq \cdots \geq w_\mu \geq 0). \tag{9}$$

The specific expression for w_i is provided in Appendix A of Ref. [8]. In this study, $\mu = \lambda/2$ is used.

Step 3: The step size $\sigma^{(g)}$ used to search for the $(g+1)$th generation is obtained by updating the previous step size $\sigma^{(g-1)}$:

$$\sigma^{(g)} = \sigma^{(g-1)} \exp\left[\frac{c_\sigma}{d_\sigma}\left(\frac{\|p_\sigma^{(g)}\|}{E\|\mathcal{N}(0,I)\|}\right) - 1\right]. \tag{10}$$

For the specific forms of c_σ, d_σ, $p_\sigma^{(g)}$, and $E\|\mathcal{N}(0,I)\|$, see Appendix A in Ref. [8]. Next, we construct a covariance matrix C that characterizes the population distribution in each generation. The $C^{(g)}$ used for generation $g+1$ is obtained by updating $C^{(g-1)}$. Because the input space for new individuals can be changed adaptively through $\sigma^{(g)}$ only, near-optimal solutions are obtained even without $C^{(g)}$. However, an isotropic search without $C^{(g)}$ is inefficient because the sphere space increases exponentially in proportion to the number of input dimensions. Thus the isotropic search becomes inefficient, particularly when the sensitivity to the objective function differs significantly among the input parameters. The covariance matrix is updated such that the input space expands in the direction in which the sensitivity of the objective function increases. A detailed discussion regarding updates to the covariance matrix is provided in Sec. 3 of Ref. [8].

The covariance matrix $C^{(0)}$ is initialized with an identity matrix I. The user should set the normal

distribution center $m^{(0)}$ and step size $\sigma^{(0)}$ based on the problem to be addressed.

III. APPLICATION TO BEAM-CHARGE MAXIMIZATION

A. Experimental setup at KEK Linac

This section describes the properties and layout of the pulsed steering magnets and beam position monitors (BPMs) used in the beam-tuning experiments at the KEK Linac.

Figure 1 shows a schematic illustration of the Linac. First, an electron beam is generated using electron guns, i.e., a thermionic dc gun and a photocathode rf gun [11]. The electron beam used to generate positrons in the beam-tuning experiments, which is known as the KBP beam, is generated by a thermionic dc gun suitable for generating high-charge beams. After passing through the bunchers [11], the electron beam enters the straight A–B sectors, followed by the arc sector known as the R sector. After turning around in the R sector, the 1.5 GeV electron beam entering the C sector is further accelerated to 3.3 GeV and strikes a positron-generating target composed of tungsten in sector 1 [12]. Positrons generated via multiple scattering in the tungsten target and the subsequent electron-positron pair generation are focused forward by the flux concentrator [13]. The positron beam is accelerated to 1.1 GeV and injected into the damping ring [14] from sector 2. During the 40 ms storage in the damping ring, the emittance is reduced via radiation damping. In the switchyard downstream of sector 5, the positron beam is injected into the beam transport line, thus resulting in a positron ring of the SuperKEKB electron-positron collider [11] (denoted as LER in Fig. 1).

The beam-tuning experiment for a single-objective optimization aims to optimize the current applied to the coils of the pulsed steering magnet [15] and consequently maximize the electron-beam charge arriving at the tungsten target. Such charge-maximization tunings are typically performed manually by operation experts. The current experiment was conducted to determine whether an optimization program can replace expert-based tunings. Six pulsed steering magnets were used in the beam-tuning experiments. Two of the pulsed steering magnets were PX(Y)_A4_4 (X is horizontal, Y is vertical) at the A sector end, and four were PX(Y)_R0_01 and PX(Y)_R0_02 near the entrance of the R sector, thus totaling six steering magnets. PX(Y)_A4_4, PX(Y)_R0_01, and PX(Y)_R0_02 are six pulsed steering magnets installed from the A sector end to the R sector end. A total of 14 BPMs [16,17] for the beam-charge measurements, each with a resolution of approximately 1%, were selected from downstream of the R sector to immediately before the tungsten target.

We set the applied current and acquired the measured charge using the EPICS protocol [18]. Because the electron-beam repetition rate during the beam experiment was 1 Hz, a wait time of 1 s was allowed after changing the applied current until the change in the applied current was reflected in the beam-orbit modification. The measured charges were averaged for all BPMs every second, and the operation was repeated two more times and averaged (which required 3 s) to obtain a better charge-measurement resolution.

B. Experimental results

This section presents the results of the beam-tuning experiments conducted at the Linac in June 2023.

Figure 2 shows the results obtained using Bayesian optimization based on the BoTorch algorithm. Panel (a) shows the peak hold values of the electron-beam charge, which varied from the 1st to the 100th step. The five solid lines (each referred to as a run) represent the cases in which the optimization parameters and applied currents of the coils are set randomly within the configuration parameter domain during initialization. We define the configuration parameter domain by the range between the minimum and maximum applied current values for each pulsed steering magnet. We choose the minimum and maximum applied current values so that unusual beam orbit does not cause severe beam loss. Hereafter, these runs are referred to as *cold starting*. In cold starting runs, initialization was performed 10 times for the first to tenth steps, which was necessary to obtain an estimation of the probability

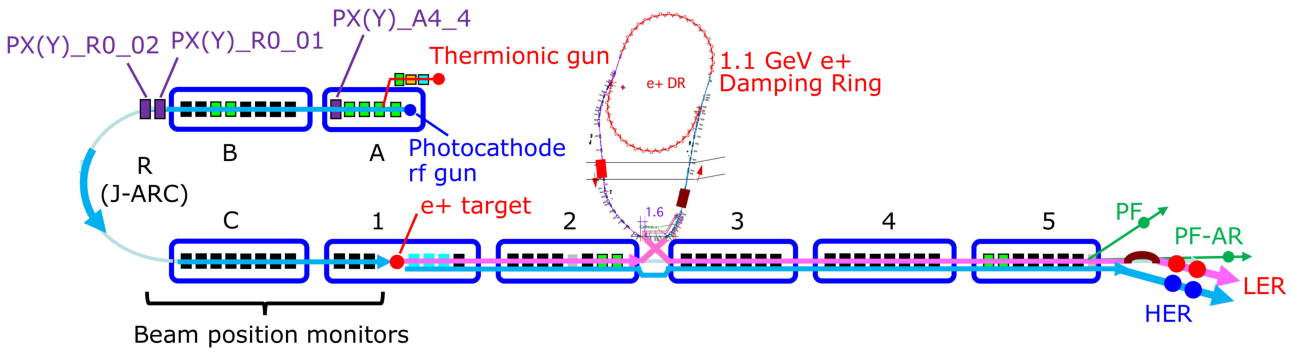

FIG. 1. Layout of pulsed steering magnets and beam position monitors at KEK Linac.

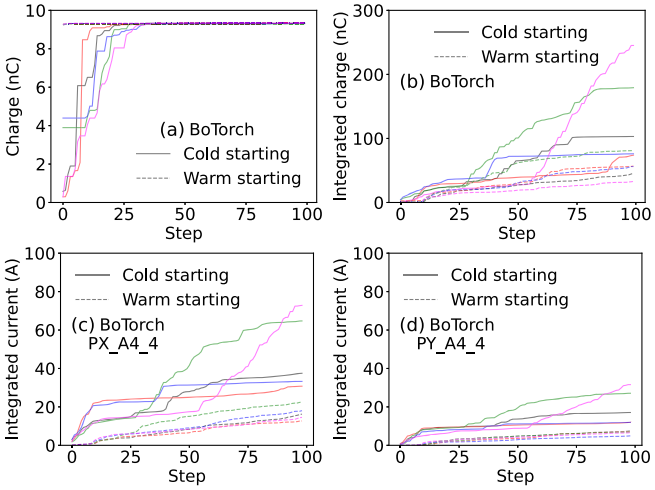

FIG. 2. Results obtained using Bayesian optimization based on BoTorch algorithm. Panel (a) indicates peak hold values of electron-beam charge. Panel (b) indicates integrated charge over steps. Panels (c) and (d) indicate integrated applied current for PX_A4_4 and PY_A4_4, respectively.

density distribution over the configuration domain. The initialization was immediately interleaved after the optimization program was started. One run for every 100 steps required approximately 20 min, which constituted primarily the wait time required to average the charge information from the BPMs. In two of the five runs, a charge of approximately 4 nC was obtained in the first step immediately after initialization. In all five runs, a maximum electron-beam charge exceeding 9.3 nC was reached in approximately 35 steps (which required 7 min), which was comparable to the pre-experiment manual-adjustment results by the experts.

The five dashed lines (which cannot be distinguished easily because they almost overlap) represent the case in which a combination of applied currents known in advance to provide a high beam charge is enqueued as the initial configuration. Hereafter, we refer to these runs as *warm starting* [19]. The enqueued applied current combinations were copied from the 10 combinations with the highest-ranking beam charges extracted from one of the cold starting runs (solid lines). Consequently, the dashed lines indicate a high peak charge immediately after initialization. The warm-starting method should benefit actual accelerator tuning provided that a certain degree of reproducibility can be guaranteed.

Panel (b) shows the integrated charge over steps, which is expressed as

$$Q_{\mathrm{int}}(t) = \sum_{i=2}^{t} |q(i) - q(i-1)|, \tag{11}$$

where $q(i)$ denotes the beam charge obtained in the ith step. Similar to panel (b), panels (c) and (d) show the integrated applied current for PX_A4_4 and PY_A4_4, respectively, which is expressed as

$$I_{\mathrm{int}}(t) = \sum_{i=2}^{t} |I(i) - I(i-1)|. \tag{12}$$

Here, $I(i)$ indicates the current applied in the ith step. The integrated charge and applied current determine the step at which the beam charge becomes steeper or milder. The five solid lines in panels (b)–(d) correspond to the cold-starting runs, where three ascended considerably from approximately the 30th to the 40th step. Comparing the three solid lines in panel (a), we discovered that the optimization shifted from exploitation to exploration around the step when the beam charge reached its maximum. Similarly, the solid magenta line in panel (a), which reached the maximum beam charge the slowest, indicated a significant shift to exploration at approximately the 55th step, as shown in panels (b)–(d). In the five dashed lines for warm starting in panel (b), the increase in the integrated charge remained gradual after the 20th step, which is consistent with the fact that the BoTorch algorithm continued to exploit the near-optimum applied current, as shown in panels (c) and (d).

Figure 3 shows the results of the TPE algorithm. We performed a beam-tuning experiment using BoTorch on June 2, 2023, 11 am–3 pm, and an experiment using the TPE on the same day, 3–6 pm; the change in the Linac conditions between the two was negligible.

The five runs indicated by the solid line in panel (a) reached 9 nC around the 40th step and the maximum charge around the 60th step. The maximum charge was slightly lower than 9.2 nC, which was in fact 0.1 nC lower than the result achieved by BoTorch [see Fig. 2(a)]. The dashed

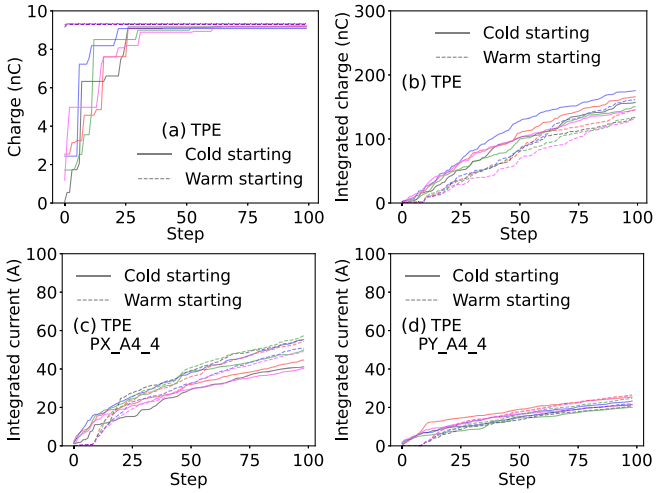

FIG. 3. Results obtained using TPE algorithm. Panel (a) indicates peak hold values of electron-beam charge. Panel (b) indicates integrated charge over steps. Panels (c) and (d) indicate integrated applied current for PX_A4_4 and PY_A4_4, respectively.

lines in panel (a) show the cases of "warm starting." Ten combinations of the applied currents were obtained from the cold-starting BoTorch results shown in Fig. 2(a). As expected from the superior combinations, the maximum charge of the warm-starting runs in Fig. 3(a) exceeded 9.3 nC.

Panel (b) shows that the integrated charge obtained by the cold-starting TPE was generally higher than that of BoTorch, as shown in Fig. 2(b), regardless of whether cold or warm starting was used. Based on panels (b)–(d), the TPE algorithm might focus more on exploration than exploitation and thus reach its maximum charge slower than BoTorch. In panel (b), we observed that up to the 50th step, the integrated charge for the warm-starting runs was slightly smaller than that for cold-starting runs; however, after the 50th step, the difference between the two became less prominent. This transition may be because, as shown in panels (c) and (d), the warm-starting TPE algorithm searches a broader range of configuration parameter domains than the cold-starting TPE after initialization. In panels (c) and (d), we observed that the integrated applied current of warm starting exceeded that of cold starting around the 30–40th steps.

Figure 4 shows the experimental results obtained using the CMA-ES. The measurements were performed on June 12, i.e., 10 days after the BoTorch measurements, as shown in Fig. 2, and the TPE measurements shown in Fig. 3. During those 10 days, the situation downstream of the electron-beam direction changed, and the beam charge arriving at the most upstream pulsed steering magnet, PX(Y)_A4_4, decreased. Therefore, the maximum charge obtained via the CMA-ES was lower, i.e., below 9 nC. In panel (a), the solid lines for the cold-starting case shows

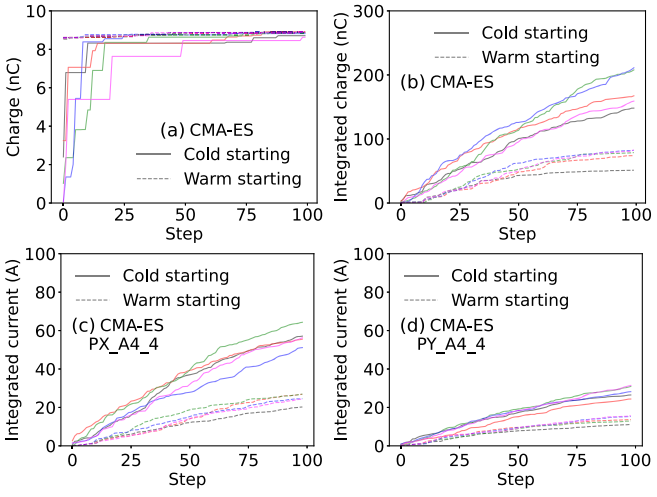

FIG. 4. Results obtained using CMA-ES algorithm. Panel (a) indicates peak hold values of electron-beam charge. Panel (b) indicates integrated charge over steps. Panels (c) and (d) show integrated applied currents for PX_A4_4 and PY_A4_4, respectively.

the maximum charge increasing the most slowly among the three algorithms considered. The dashed lines show the warm-starting case, where the 10 combinations of the applied currents were obtained from the cold-starting CMA-ES measurements (i.e., one of the solid lines). As shown in panels (b)–(d), the integrated charge and applied current were consistently lower for warm starting than for cold starting. This tendency indicates that, as with BoTorch, in the case of warm starting, the optimized applied current is enqueued as the initial value combination and the surrounding domain of that combination is exploited.

The three algorithms, shown in Figs. 2–4, were terminated after 100 steps. However, for BoTorch and the TPE, only one run each was tested, and the optimization was extended to 300 steps. Based on the results, BoTorch continued to be optimized, thus emphasizing exploitation near the maximum charge. However, the TPE shifted from exploitation to exploration after the 150th step. Systematic measurements based on a significantly higher number of runs shall be attempted in future studies.

Figure 5 shows the empirical distribution function (EDF), which is defined as

$$\mathrm{EDF}(q_{\mathrm{thr}}) = \frac{\text{number of steps of } q < q_{\mathrm{thr}}}{100} = \frac{1}{100}\sum_{i=1}^{100} \mathbb{1}_{q<q_{\mathrm{thr}}}. \tag{13}$$

Panel (a) shows the results of the BoTorch algorithm averaged over five runs for the cold-starting (solid line) and warm-starting (dashed line) cases. Each line shows a steep increase in the EDF beginning at approximately 8.5 nC, thus indicating that many steps yielded a beam charge exceeding 8.5 nC. For cold starting, approximately

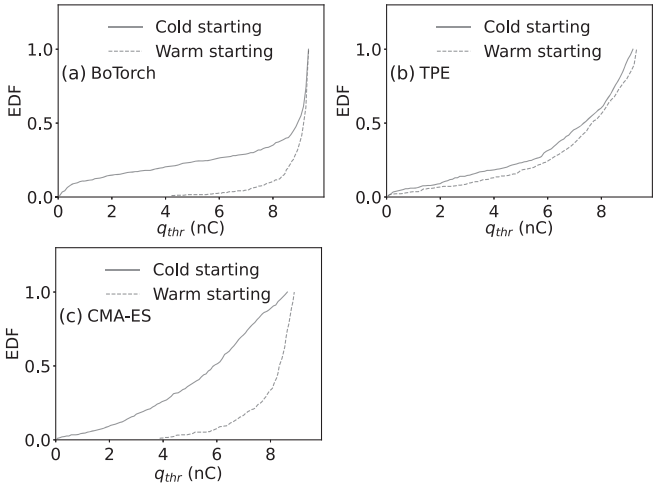

FIG. 5. Empirical distribution functions (EDF) averaged over five runs. Panels (a)–(c) show results of BoTorch, TPE, and CMA-ES, respectively.

60% of the steps were distributed above 8.5 nC, whereas for warm starting, approximately 80% of the steps were distributed above 8.5 nC. The results shown in panel (a) are consistent with the superior performance of BoTorch shown in Fig. 2.

Panel (b) shows the results of the TPE measured on the same day as that of BoTorch. For a specified beam charge, the warm-starting TPE (dashed line) consistently showed a lower EDF than the cold-starting TPE (solid line), thus indicating that the warm-starting steps were distributed at a slightly higher beam charge. However, the shapes of the EDFs were almost identical, and the differences were insignificant compared with those of BoTorch and the CMA-ES. Slight differences between the two curves suggest that the warm-starting TPE affects the maximum charge through initialization but negligibly affects the trade-off between exploration and exploitation. This trend is consistent with the integrated charge and applied current distributions shown in Figs. 3(b)–3(d). As noted in Sec. II B, the trade-off between exploration and exploitation depends on the parameter γ. Therefore, the difference between warm-starting and cold-starting TPEs can be significant if we choose γ other than 0.1 used for this study.

Panel (c) shows the measurement results obtained using the CMA-ES, where the EDF integrated from 0 to 8 nC for cold starting was the largest among the three algorithms considered. Similar to the results shown in Fig. 4(a), the CMA-ES algorithm for cold starting performed an exploration-oriented optimization in this experiment. However, for warm starting, more than 70% of the steps occurred at 8 nC or higher, thus indicating that the optimization focused on exploitation. The trends in panel (c) are similarly shown in Figs. 4(b)–4(d) for warm starting (dashed line).

Figure 6 shows the importance of each parameter, namely, the effectiveness of the applied currents in increasing the beam charge. The importance of the parameters can be quantified using a method proposed in Ref. [20], which is based on a random forest prediction model [21] and functional analysis of variance (fANOVA). First, a random-forest model was established to predict the average algorithm performance over the configuration domain. Subsequently, the fANOVA decomposes the variance of the overall algorithm performance into additive components, with each corresponding to a subset of the algorithm parameters. Finally, the fraction of variance associated with each subset of parameters relative to the overall performance variance quantifies the importance of the corresponding subset.

Panel (a) shows the results of the BoTorch measurements. The black circles and red squares represent the average of five cold- and warm-starting runs, respectively. For cold starting, the maximum importance was ∼0.5 for pulsed steering magnet PX_A4_4. Because the importance was normalized such that the sum was 1, PX_A4_4 alone appeared to have contributed to approximately 50% of the

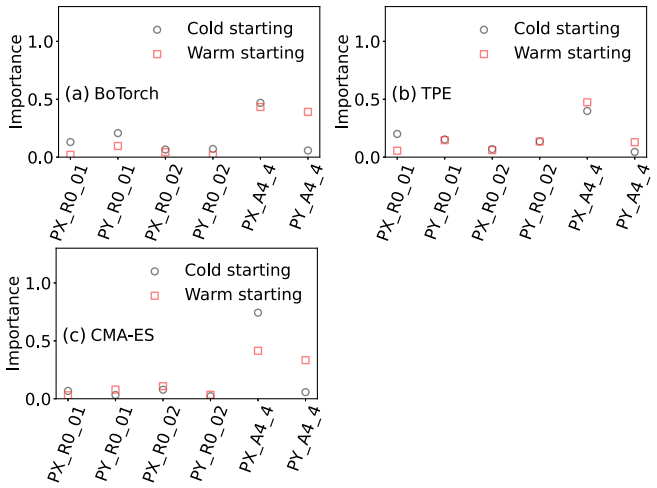

FIG. 6. Importance of each input parameter averaged over five runs. Panels (a), (b), and (c) show results of BoTorch, TPE, and CMA-ES, respectively.

importance. As shown in Fig. 1, PX_A4_4 was the most upstream of the three horizontal magnets used in the beam-tuning experiment. Therefore, PX_A4_4 is expected to exert the most significant effect on the horizontal orbit modification, thus resulting in a higher importance for the objective function (i.e., beam charge). Because of a sizable orbit error in the horizontal direction, the beam established contact with the beam collimator in the arc R sector, thus resulting in a significant beam-charge loss. This may explain the higher importance of the horizontal direction PX_A4_4 compared with that of the vertical direction PY_A4_4. For warm starting, the importance of PY_A4_4 increased to 0.4. As shown by the integrated charge in Fig. 2(b), the integrated applied current in Figs. 2(c) and 2(d), and the EDF in Fig. 5(a), most of the 100 parameter sets for warm starting did not change significantly from the best parameter set that yielded the maximum charge. That is, the applied current of PY_A4_4, which was located upstream, was almost optimized immediately after the initialization. Thus a slight change in the applied current significantly affected the beam charge loss and became more important. In two locations in the R sector, the vertical beta function exceeded 100m. In addition, an electron beam generated by the thermionic dc gun indicated a large emittance. Therefore, if the beam orbit is shifted vertically, then a portion of the bunch with a large transverse size hits the beam pipe, thus resulting in beam-charge loss.

Panel (b) presents the TPE results. The results of cold and warm starting were similar. As shown in Figs. 3(b)–3(d) and Fig. 5(b), the trade-off between exploitation and exploration changed only slightly, regardless of whether cold or warm starting was used in TPE. Thus, we can assume that the importance of each parameter is similarly distributed for both the cold- and warm-starting runs. Because the TPE optimizes

with emphasis on exploration even for warm starting, the results achieved are comparable to those yielded by BoTorch for cold starting.

Panel (c) presents the CMA-ES results. The results for warm starting were comparable to those of BoTorch, which is as expected owing to the similarity of the EDFs shown in Figs. 5(a) and 5(c). Meanwhile, the importance of PX_A4_4 for cold starting exceeded 0.7 for the CMA-ES, as compared with 0.5 for BoTorch. Although we have yet to achieve quantitative understanding, we hypothesize that the anticorrelation between the more critical PX_A4_4 and less critical PX_R0_01 may affect the importance of PX_A4_4. Based on panels (a) and (b), the importance of PX_R0_01 is 0.15–0.2, whereas it is less than 0.1 in panel (c).

Figure 7 shows the importance of the two-parameter combinations. The results yielded by the warm-starting BoTorch and warm-starting TPE are shown above and below the diagonal line, respectively. For simplicity, we set the values on the diagonal line to zero. In the BoTorch results, the combination of PX_A4_4 and PY_A4_4 showed the highest importance of ∼0.14. Combining the results for warm starting (red squares) in Fig. 6(a), we observed that the order of importance from highest to lowest was PX_A4_4, PY_A4_4, and the combination of both. Meanwhile, the importance of the other pulsed steering magnets and their combinations was less than 0.1.

The TPE results below the diagonal line show that the importance for the combination of each parameter was 0.05 at the maximum. PY_A4_4 was less critical in the TPE, even for warm starting, as shown in Fig. 6(b). Consequently, the combination of PX_A4_4 and PY_A4_4 was less critical.

Figures 6 and 7 show that the applied currents of PX_A4_4 and PY_A4_4, which were the most upstream pulsed steering magnets used in the experiment, functioned similarly as the other steering magnets for the three algorithms.

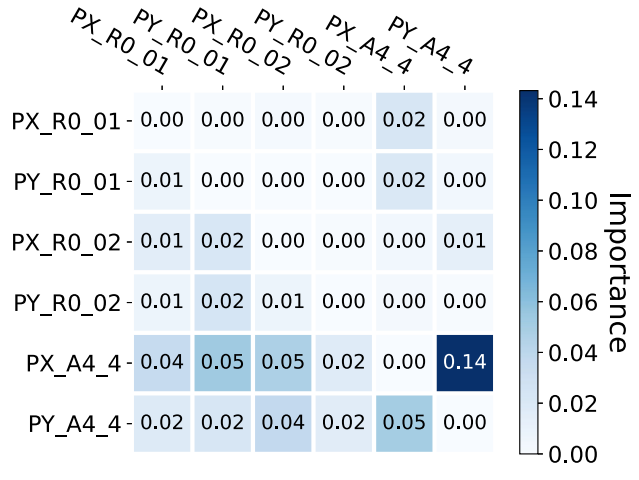

FIG. 7. Importance of each input parameter combination. Numbers shown above and below diagonal line indicate BoTorch and TPE results, respectively. Both results are for warm starting.

Therefore, only the applied currents of PX_A4_4 and PY_A4_4 appeared to be sufficient for the enqueued initial parameter values considered under warm starting.

In the next beam-tuning experiment, we plan to perform beam tuning at different sectors of Linac using other magnets and other types of objective functions to obtain more general insights into machine learning-assisted beam tuning.

IV. MULTIOBJECTIVE OPTIMIZATION

This section provides an overview of the methodology of multiobjective optimization, i.e., the case in which multiple objective functions are optimized simultaneously. Attempts to apply multiobjective optimization to accelerator control have already been made, for example, in the Argonne wakefield accelerator photoinjector [22,23], and in SLAC MeV-UED, part of the LCLS user facility [24].

The M-dimensional multiple-objective functions are denoted as $f^{(1)}, f^{(1)}, \ldots, f^{(M)}$. In the beam-tuning experiment, we set $M = 2$ to accommodate the two objective functions (maximization of the beam charge and minimization of the dispersion function). For simplicity, we assume that the goal of multiobjective optimization is to maximize all dimensions.

The simultaneous optimization of multiobjective functions can be redefined as obtaining all the Pareto optimal solutions. To illustrate the Pareto optimal solution, we define the dominance relation. For the two objective functions $\boldsymbol{f}(\boldsymbol{x})$ and $\boldsymbol{f}(\boldsymbol{x}')$, the relation

$$\boldsymbol{f}(\boldsymbol{x}) \succeq \boldsymbol{f}(\boldsymbol{x}') \Leftrightarrow \forall\, m \in \{1, \ldots, M\} f^{(m)}(\boldsymbol{x}) \geqq f^{(m)}(\boldsymbol{x}') \quad (14)$$

indicates that $\boldsymbol{f}(\boldsymbol{x})$ dominates $\boldsymbol{f}(\boldsymbol{x}')$ if $f^{(m)}(\boldsymbol{x})$ is greater than or equal to $f^{(m)}(\boldsymbol{x}')$ for all dimensions. Here, $\boldsymbol{x}_i$ represent the input variables (applied currents of the pulsed steering magnets) and $\boldsymbol{f}_i$ the objective functions (a beam charge and a dispersion function). The last inequality is presented in Eq. (14) as we are addressing a maximization problem. For example, as shown in Fig. 8, $\boldsymbol{f}_+$ dominates $\boldsymbol{f}_4$. We regard $\boldsymbol{f}$ as a Pareto-optimal solution when no other point in the objective function space dominates $\boldsymbol{f}$. Generally, more than one Pareto-optimal solution exists for multiple objective functions.

The Pareto front is the surface created when plotting the Pareto solution set. Multiobjective optimization aims to efficiently obtain many Pareto-optimal solutions near the Pareto front by discounting the superiority or inferiority of the multiple Pareto-optimal solutions. To obtain the Pareto front, we introduce a Pareto hypervolume. Let $\mathcal{D} = \{(\boldsymbol{x}_i, \boldsymbol{f}_i)\}_{i=1}^{N}$ denote the current dataset, where N is the dataset size. The Pareto front in the dataset $\mathcal{D}$ expands with each data addition.

Once a reference point is determined to evaluate the expansion, a hyper-rectangle can be defined using the reference point $\boldsymbol{r}$ and the Pareto solution set $\mathcal{P} \in \mathcal{D}$:

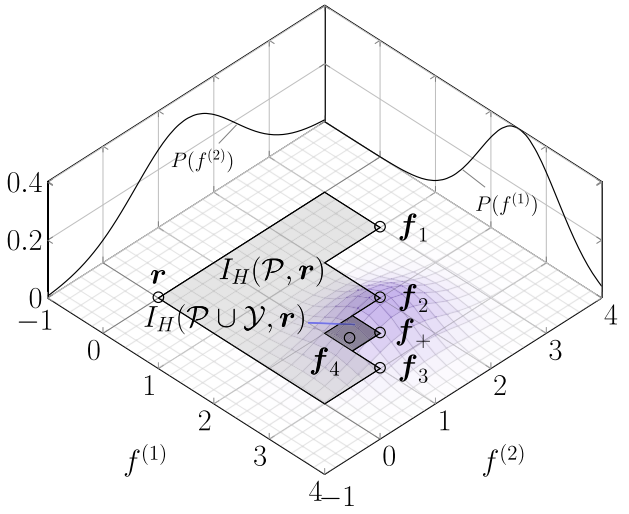

FIG. 8. Example of hypervolume indicator I_H with Pareto solution set $\{(\boldsymbol{x}_i,\boldsymbol{f}_i)\}_{i=1}^3$.

$$\cup_{i=1}^{\mathcal{P}}\,[\boldsymbol{r},\boldsymbol{f}_i]. \tag{15}$$

The hypervolume indicator in Eq. (15) is the M-dimensional Lebesgue measure, which is expressed as

$$I_H(\mathcal{P},\boldsymbol{r}) = \lambda_M(\cup_{i=1}^{\mathcal{P}}\,[\boldsymbol{r},\boldsymbol{f}_i]). \tag{16}$$

The shaded light-gray area in Fig. 8 shows the hypervolume indicator with the Pareto solution set $\mathcal{P} = \{(\boldsymbol{x}_i,\boldsymbol{f}_i)\}_{i=1}^3$, and reference point $\boldsymbol{r}$. The hypervolume indicator increases monotonically for each additional data point. The shaded dark-gray area shows an increase in the hypervolume indicator owing to new observations $\mathcal{Y} = (\boldsymbol{x}_+,\boldsymbol{f}_+)$. Based on this increase, we can define an acquisition function, i.e., the expected hypervolume improvement (EHVI), which is the expected increment in the hypervolume indicator before and after obtaining a new observation $\mathcal{Y}$

$$\mathrm{EHVI}(\mathcal{Y},\mathcal{P},\boldsymbol{r}) = \int [I_H(\mathcal{P}\cup\mathcal{Y},\boldsymbol{r}) - I_H(\mathcal{P},\boldsymbol{r})]\,p(\boldsymbol{f}_+|\boldsymbol{x}_+,\mathcal{D})d\boldsymbol{f}_+. \tag{17}$$

The posterior $p(\boldsymbol{f}_+|\boldsymbol{x}_+,\mathcal{D})$, which is the distribution indicated by the blue surface in Fig. 8, is approximated via a Gaussian process in the Bayesian optimization. Analogous to the single-objective optimization, as discussed in Sec. II, the location $\boldsymbol{x}$ that maximizes the acquisition function EHVI is selected and then input to the objective function to advance the Pareto front. The methodology described above assumes a multiobjective Bayesian optimization. A detailed description of this multi-objective optimization based on the TPE algorithm is provided in Refs. [25,26].

V. APPLICATION TO BEAM CHARGE AND DISPERSION SIMULTANEOUS OPTIMIZATION

A. Experimental setup at KEK Linac

To investigate the feasibility of applying multiobjective Bayesian optimization to accelerator tuning, we attempted to simultaneously maximize the electron-beam charge and minimize the dispersion function.

The unexpected dispersion function in the Linac and beam-transport line increases the emittance of the injection beam and further reduces the injection efficiency to the light source storage rings (PF and PF-AR) and SuperKEKB [11] downstream of the beam-transport line. Therefore, the beam must be adjusted such that the sizable dispersion function in sector R (see Fig. 1) does not leak downstream.

In this beam-tuning experiment, we used an electron beam generated by a thermionic dc gun dedicated for positron generation. Notably, the dispersion function in sector 1 does not significantly affect positron generation in the actual operation. We attempted to minimize the dispersion function in this experiment such that better electron beams would be received by the PF, PF-AR, and SuperKEKB. Notably, the electron beam generated by the rf electron gun was supplied to the PF and PF-AR when this experiment was conducted and was not used in this study.

In this study, the dispersion function was not used as an objective function. Instead, the following dispersion-position function was used for simplicity, which multiplies the square of the dispersion function by the sum of the squares of the horizontal and vertical positions:

$$f_{\text{disp-pos}} = \sum_{i=1}^{14}(\eta_{x,i}^2+\eta_{y,i}^2)\sum_{j=1}^{14}(d_{x,j}^2+d_{y,j}^2). \tag{18}$$

The dispersion functions and beam positions were measured using 14 BPMs. The product of the dispersion function and position was adopted to simultaneously reduce both the dispersion function and beam-orbit residual. The dispersion function at each BPM location was measured using the inevitably occurring beam-energy jitter. Compared with the case where the dispersion function is measured by intentionally changing the energy-adjustment knob, the method using energy jitter enables measurements to be performed while the beam is being supplied to the light source storage rings or SuperKEKB because the adjustment knob is fixed. However, the energy-jitter method relies on randomly generated jitter and requires a long measurement time to obtain sufficient resolution. In this experiment, 100 data points were required after the pulsed magnet settings were changed. Because the beam repetition rate was 1 Hz, a waiting period of 100 s was permitted after the magnet settings were changed.

B. Experimental results

Figure 9 shows the scatter plots of the obtained beam charge vs the dispersion-position function in Eq. (18) for each step. Measurements were performed on June 1, 2023, 2–4 am, with 100 steps performed using the BoTorch algorithm. The green squares indicate the 10 steps performed until the end of initialization, the blue open circles indicate the 11th to 100th steps, and the red dots are the Pareto-optimal solution set (five points in total). The initialization explores the beam charge and dispersion-position function domain more extensively than the pairs obtained after the initialization. The beam charges were generally distributed above 6 nC, and the charge optimization was efficient. Meanwhile, the dispersion-position function exhibited a tail exceeding 2 $m^2\,mm^2$, which can be further improved. Although not used in this study, OPTUNA implements a constrained optimization function. If either the dispersion function or beam-orbit position is constrained, then the optimization can focus of regions where the dispersion-position function is small, e.g., smaller than 1 $m^2\,mm^2$.

Figure 10 shows a scatter plot of the beam charge vs dispersion-position function obtained using the TPE algorithm via 200 steps. The result indicates an exploration-oriented optimization of both the beam charge and dispersion-position function, unlike the BoTorch result presented in Fig. 9. This trend is consistent with TPE's focus on exploration instead of exploitation, as discussed in Sec. III B. The TPE algorithm data were obtained on June 12, 2023, 6–10 pm; therefore, the beam conditions may have changed since June 1 when we conducted the beam test using the BoTorch algorithm.

As shown in Figs. 9 and 10, we obtain Pareto-optimal solutions for both algorithms. Only a few Pareto-optimal solutions were available for 100 or 200 steps, thus clearly indicating a low-cost performance in terms of the beam-time. The Pareto-optimal solution must be obtained promptly for time-consuming measurements, e.g., more than 60 s per measurement, as in the current tuning experiment. In the future, we plan to test the efficiency of obtaining Pareto-optimal solutions using a constrained optimization algorithm.

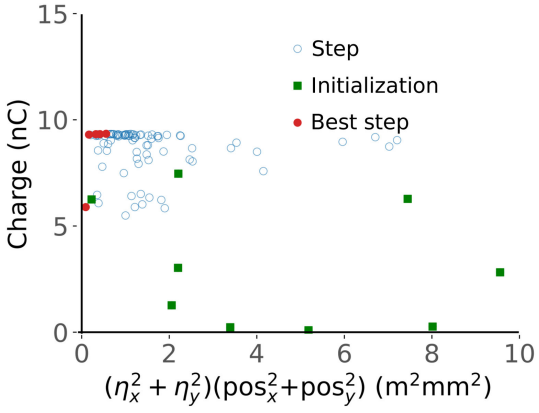

FIG. 9. BoTorch result for multiobjective optimization on beam charge (vertical axis) and dispersion-position function (horizontal axis).

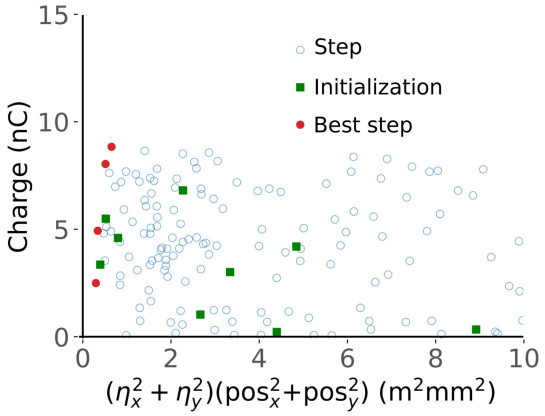

FIG. 10. TPE result for multiobjective optimization on beam charge (vertical axis) and dispersion-position function (horizontal axis).

VI. CONCLUSIONS

We conducted beam-tuning experiments at the KEK Linac using Bayesian optimization (BoTorch), a TPE, and the CMA-ES to determine the feasibility of using machine learning, in particular optimization algorithms.

In a single-objective optimization experiment to maximize the electron-beam charge from sector C to sector 1 of the Linac, the beam orbit was adjusted by optimizing the applied current of six pulsed steering magnets. The maximum beam charge obtained in the beam-tuning experiment was comparable to that obtained in expert-based tunings. Approximately 35 steps (10 of which pertained to initialization) were required to reach the maximum beam charge when using the cold-starting BoTorch. On average, the TPE and CMA-ES achieved a lower beam charge under cold starting compared with BoTorch, even after 100 steps. Under warm starting, BoTorch and the CMA-ES showed excellent optimization performance from the initialization phase, where the enqueued initial values were utilized; however, the optimization performance of the TPE under warm starting did not differ significantly from that of cold starting. We conclude that the optimization algorithms proposed in this study can replace manual tuning by experts for beam-charge maximization using steering magnets.

Multiobjective optimization was tested to simultaneously maximize the beam charge and minimize the dispersion-position function. The multiobjective optimization task was to obtain as many effective Pareto-optimal solutions as feasible. The dispersion function was measured using the inevitable beam energy jitter. Two algorithms,

i.e., BoTorch and the TPE, were used, and their results were compared. As shown in the single-objective optimization results, the beam charge vs dispersion-position function distribution was exploitation-oriented for BoTorch and exploration-oriented for the TPE. Both algorithms yielded four to five Pareto-optimal solutions over 100–200 steps. The efficiency of obtaining the optimal solution is essential for applying multiobjective optimization to accelerator tuning, where each step is time consuming and hence expensive. In the next test, we shall introduce a constrained optimization algorithm to improve the efficiency of obtaining the optimal solution. In addition, the number of parameters shall be increased to approximately 20 to assess the applicability of multivariable optimization to beam tuning.

The machine-learning-based beam-tuning tool developed for this experiment will be applied to the beam tuning of SuperKEKB (e.g., beam injection from the beam-transport line to the main ring, correction for horizontal and vertical couplings, and adjustment of the beam collimator head position) [27,28].

ACKNOWLEDGMENTS

The authors would like to thank the members of the KEK Linac commissioning group and the KEK Linac operators for their cooperation and assistance with our beam study. Additionally, the authors thank the Belle II machine-detector interface group for their cooperation and fruitful discussions.

[1] J. Duris, D. Kennedy, A. Hanuka, J. Shtalenkova, A. Edelen, P. Baxevanis, A. Egger, T. Cope, M. McIntire, S. Ermon, and D. Ratner, Bayesian optimization of a free-electron laser, Phys. Rev. Lett. **124**, 124801 (2020).

[2] R. Roussel *et al.*, Bayesian optimization algorithms for accelerator physics, arXiv:2312.05667.

[3] R. Garnett, *Bayesian Optimization* (Cambridge University Press, Cambridge, England, 2023).

[4] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning* (The MIT Press, Cambridge, MA, 2005).

[5] J. O. Berger, *Statistical Decision Theory and Bayesian Analysis*, Springer Series in Statistics 2nd ed. (Springer, New York, 1985).

[6] J. Bergstra, R. Bardenet, Y. Bengio, and B. Kégl, Algorithms for hyper-parameter optimization, in *Proceedings of the 24th International Conference on Neural Information Processing Systems, NIPS-2011, Granada, Spain* (Curran Associates Inc., Red Hook, NY, 2011), pp. 2546–2554.

[7] S. Watanabe, Tree-structured parzen estimator: Understanding its algorithm components and their roles for better empirical performance, arXiv:2304.11127.

[8] N. Hansen, The CMA evolution strategy: A tutorial, arXiv:1604.00772.

[9] T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama, Optuna: A next-generation hyperparameter optimization framework, in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD-2019, Anchorage, AK* (Association for Computing Machinery, New York, 2019), pp. 2623–2631.

[10] M. Balandat, B. Karrer, D. R. Jiang, S. Daulton, B. Letham, A. G. Wilson, and E. Bakshy, BoTorch: A framework for efficient Monte-Carlo Bayesian optimization, in *Proceedings of the Annual Conference on Advances in Neural Information Processing Systems 33, NeurIPS 2020* (Curran Associates Inc., Red Hook NY, 2020).

[11] Y. Funakoshi and K. Furukawa, SuperKEKB Design Report, KEK, 2020, https://www-linac.kek.jp/linac-com/report/skb-tdr/.

[12] T. Kamitani *et al.*, SuperKEKB positron source construction status, in *Proceedings of the 5th International Particle Accelerator Conference, IPAC-2014, Dresden, Germany* (JACoW, Geneva, Switzerland, 2014), pp. 579–581.

[13] Y. Enomoto, K. Abe, N. Okada, and T. Takatomi, A new flux concentrator made of Cu alloy for the SuperKEKB positron source, in *Proceedings of the 12th International Particle Accelerator Conference, IPAC-2021, Campinas, Brazil*, (JACoW, Geneva, Switzerland, 2021), pp. 2954–2956, 10.18429/JACoW-IPAC2021-WEPAB144.

[14] N. Iida *et al.*, Commissioning of positron damping ring and the beam transport for SuperKEKB, in *Proceedings of the 62nd ICFA ABDW on High Luminosity Circular e^+e^- Colliders eeFACT-2018, Hong Kong, China* (JACoW, Geneva, Switzerland, 2019), pp. 152–156, 10.18429/JACoW-eeFACT2018-TUPAB07.

[15] Y. Enomoto *et al.*, Pulse-to-pulse beam modulation for 4 storage rings with 64 pulsed magnets, in *Proceedings of the 29th Linear Accelerator Conference, LINAC-2018, Beijing, China* (JACoW, Geneva, Switzerland, 2019), pp. 609–614, 10.18429/JACoW-LINAC2018-WE1A06.

[16] M. Satoh *et al.*, Synchronized beam position measurement for SuperKEKB Injector Linac, in *Proceedings of the 9th International Particle Accelerator Conference, IPAC-2018, Vancouver, BC, Canada* (JACoW, Geneva, Switzerland, 2018), pp. 4159–4162, 10.18429/JACoW-IPAC2018-THPMF045.

[17] F. Miyahara, K. Furukawa, M. Satoh, Y. Seimiya, and T. Suwada, Operational performance of new detection electronics for stripline-type beam position monitors at the SuperKEKB injector Linac, in *Proceedings of the 8th International Beam Instrumentation Conference, IBIC-2019, Malmo, Sweden* (JACoW, Geneva, Switzerland, 2019), pp. 522–525, 10.18429/JACoW-IBIC2019-WEPP006.

[18] L. R. Dalesio, J. O. Hill, M. Kraimer, S. Lewis, D. Murray, S. Hunt, W. Watson, M. Clausen, and J. Dalesio, The experimental physics and industrial control system architecture: Past, present, and future, Nucl. Instrum. Methods Phys. Res., Sect. A **352**, 179 (1994).

[19] M. Nomura, S. Watanabe, Y. Akimoto, Y. Ozaki, and M. Onishi, Warm starting CMA-ES for hyperparameter optimization, Proc. AAAI Conf. Artif. Intell. **35**, 9188 (2021).

[20] F. Hutter, H. Hoos, and K. Leyton-Brown, An efficient approach for assessing hyperparameter importance, in *Proceedings of the 31st International Conference on Machine Learning, Beijing, China*, edited by E. P. Xing and T. Jebara (PMLR, Bejing, China, 2014), Vol. 32, pp. 754–762.

[21] F. Hutter, H. H. Hoos, and K. Leyton-Brown, Sequential model-based optimization for general algorithm configuration, in *Proceedings of the 5th International Conference on Learning and Intelligent Optimization, LION-2005, Rome, Italy* (Springer-Verlag, Berlin, Heidelberg, 2011), pp. 507–523.

[22] A. Edelen, N. Neveu, M. Frey, Y. Huber, C. Mayes, and A. Adelmann, Machine learning for orders of magnitude speedup in multiobjective optimization of particle accelerator systems, Phys. Rev. Accel. Beams **23**, 044601 (2020).

[23] R. Roussel, A. Hanuka, and A. Edelen, Multiobjective Bayesian optimization for online accelerator tuning, Phys. Rev. Accel. Beams **24**, 062801 (2021).

[24] F. Ji, A. Edelen, R. England, P. Kramer, D. Luo, C. Mayes, M. Minitti, S. Miskovich, M. Mo, A. Reid, R. Roussel, X. Shen, X. Wang, and S. Weathersby, Multi-objective Bayesian optimization at SLAC MEV-UED, in *Proceedings of the 13th International Particle Accelerator Conference, IPAC-2022, Bangkok, Thailand* (JACoW, Geneva, Switzerland, 2022), pp. 995–998.

[25] Y. Ozaki, Y. Tanigaki, S. Watanabe, and M. Onishi, Multiobjective tree-structured parzen estimator for computationally expensive optimization problems, in *Proceedings of the 2020 Genetic and Evolutionary Computation Conference, GECCO 2020, Cancún, Mexico* (Association for Computing Machinery, New York, 2020), pp. 533–541.

[26] Y. Ozaki, Y. Tanigaki, S. Watanabe, M. Nomura, and M. Onishi, Multiobjective tree-structured parzen estimator, J. Artif. Intell. Res. **73**, 1209 (2022).

[27] C. Xu, T. Boltz, A. Mochihashi, A. Santamaria Garcia, M. Schuh, and A.-S. Müller, Bayesian optimization of the beam injection process into a storage ring, Phys. Rev. Accel. Beams **26**, 034601 (2023).

[28] J. Kirschner, M. Mutný, A. Krause, J. Coello de Portugal, N. Hiller, and J. Snuverink, Tuning particle accelerators with safety constraints using Bayesian optimization, Phys. Rev. Accel. Beams **25**, 062802 (2022).